Client-side Subsetting and Regridding for Intercomparison and Validation of MERRA Data

Arlindo da Silva

Global Modeling and Assimilation Office, NASA/GSFC

Arlindo.daSilva@nasa.gov

MERRA User's Workshop
AMS Annual Meeting, Phoenix, AZ
11 January 2009

Outline

- Statement of the problem
- Introduce LATS4D, a CLI tool for
  - Subsetting (variable, region, levels, time)
  - Averaging, numerical operations (finite differencing, integration, thermodynamical, math functions, etc.)
  - Reformatting (netcdf, hdf, grib, binary, etc)
  - Regridding
- Show by example, how to perform these operations on MERRA collections via OPeNDAP
  - In particular, how to make a MERRA collection impersonate another reanalysis
- Concluding Remarks

Statement of the problem

- As of yet, existing re-analyses do not share a standard convention for variable names and grid definition
- A lot of effort is wasted coping with different file formats, deciphering foreign metadata conventions
  - On-line data ordering for later delivery has gotten faster and easier but still stands in between an idea and a quick exploratory analysis (using researcher own tools.)
- By using OPeNDAP and a few simple tools we hope to empower the user to get the data they want in a few of the formats and grids they are familiar with
  - In the comfort of their own scripting language
  - Note: such tools could also be implement as a service

Introducing LATS4D

- LATS4D is a command line tool for subsetting, regridding and reformating
- Although LATS4D has GrADS as an internal backend you do not need to know GrADS to use it
- It works on any GrADS readable *file*:
  - OPeNDAP, NetCDF-3, HDF-4, NetCDF-4/HDF-5, GRIB-1, GRIB-2, and some binaries.
- Although it is quite versatile, LATS4D (like GDS) is also very limited in some respects, e.g.:
  - Can only handle GrADS readable datasets (CF-1 compliant)
  - No support for Level 2 satellite data

LATS4D, cont

- It comes as 2 files:
  - **lats4d.gs**: GrADS script
  - **lats4d.sh**: Simple shell script wrapper
    - Generally used from the command line
    - Win32 GrADS bundles in **sh.exe** (in reality, GNU bash)
- Requirements:
  - GrADS installation
    - Version 1.9-rc1 or later preferred
    - Limitted functionality with GrADS v2.0
- USB memory stick has all you need to run lats4d on Windows, Linux i686/x86_64 and Mac OS X intel/ppc

NAME

lats4d - file conversion and subsetting utility

SYNOPSIS

lats4d.sh [-nc|-hdf|-dods|-dap] option(s)

DESCRIPTION

lats4d.sh is a command line interface to GrADS and the **lats4d.gs** script. It starts the required GrADS binary depending on the (-hdf,-nc,-dods,-dap) option specified, then runs lats4d.gs, exiting from GrADS upon completion.

OPTIONS

-nc	for producing GRIB or NetCDF files
-hdf	for producing GRIB or HDF-SDS files (default)
-dods	for reading OPeNDAP URLs with gradsdods (v1.9)
-dap	for reading OPeNDAP URLs with gradsdap (v2.0) (no GRIB, NetCDF or HDF output in this case)
option(s)	for a list of **lats4d.gs** options enter: lats4d.sh -h

NAME

lats4d - LATS for Dummies (Version 2.0.0 of 7 Jan 2009)

SYNOPSIS

lats4d [-i fn] [-o fn] [-cal calendar] [-center ctr] [-de fn]
[-format fmt] [-ftype ctl|sdf|xdf] [-freq ...]
[-func expr] [-h] [-grid type]
[-lat y1 y2] [-levs ...] [-lon x1 x2]
[-model mod] [-mean] [-precision nbits] [-table tab]
[-time t1 t2 [tincr]] [-title ...]
[-v] [-vars ...] [-xvars] [-zrev] [-q]

DESCRIPTION

A minimum fuss gs script for writing NetCDF, HDF-SDS or GRIB files from GrADS using the PCMDI LATS interface (http://www-pcmdi.llnl.gov). This script can serve as a general purpose file conversion and subsetting utility. Any GrADS readable file (GrADS IEEE, GSFC Phoenix, GRIB, NetCDF or HDF-SDS) can be subset and converted to GRIB, NetCDF, HDF-SDS, flat binary (direct access) or sequential (FORTRAN) binary using a single command line. When writing binary files, the user can request the files to be little or big endian, regardless of the endianess of the hardware.

I/O OPTIONS:

-i fn — input file name; it can be any of the following:
- an ASCII control (ctl) file used for GRIB, IEEE files, and as of GrADS v1.9, for NetCDF/HDF files as well.
- a binary NetCDF file/template
- a binary HDF-SDS file/template
- an ASCII data descriptor file (ddf) used for non-COARDS compliant NetCDF/HDF-SDS files through the "xdfopen" command

-j fn — secondary input file name with same structure as the the primary input file (same variables, grid, times). This is useful for comparing files.

-o fn — output (base) file name; default: "grads.lats"

-format fmt — LATS file format; fmt can be one of the following:
- coards
- grib
- grads_grib
- sequential
- stream
- stats
- $script.gs
- $script.gsf

where $script is a generic script name.

Subsetting Options

-lat	y1 y2	latitude range, e.g., "-30 30" for 30S thru 30N; default: latitude dimension environment
-levs	lev1 ... levN	list of levels; default: all levels
-lon	x1 x2	longitude range, e.g., "-50 20" for 50W thru 20E; default: longitude dimension environment
-time	t1 t2 [tincr]	time range and time increment in units of the "delta t" in the input file; "tincr" is optional;
-vars	var1 ... varN	list of variables; default: all variables on the current file will be written to the output file

Transformation / Misc Options

-func expr	Evaluates the expression "expr" before writing to the output file. The character "@" is used to denote the variable name in "expr". Example: -func ave(@,t-1,t+1) will replace "@" with each variable name and produce a file with running means. Default: expr = @
-h	displays this man page
-mean	saves time mean to file; the actual averaging period is specified with the "-time" option; the "tincr" parameter is the time increment for the average (see GrADS ave() function)
-v	verbose mode

Easy Regridding Options

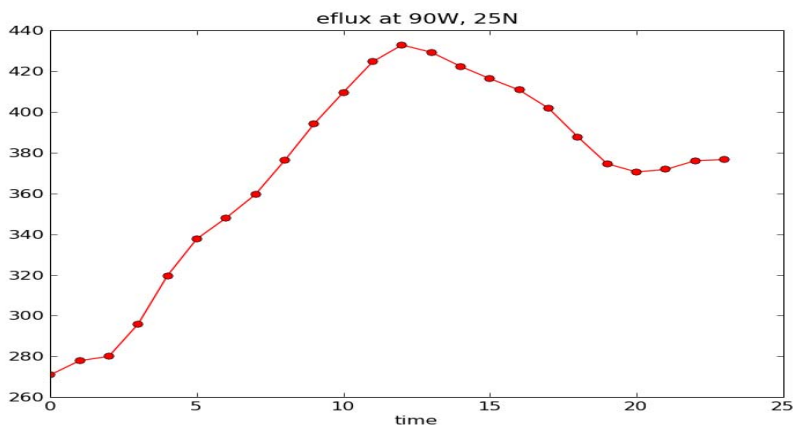
-ncept62	NCEP T62 gaussian grid
-ncep2.5	NCEP 2.5x2.5 global grid used for surface fields (Reanalysis 2): [90S,90N], [0,360)
-era2.5	Same as NCEP 2.5
-gpcp2.5	GPCP 2.5x2.5 lat/lon grid: [88.75S,875N], [1.25E,358.75] Notice that this grid differs from NCEP/ERA 2.5 grid.
-jrat160	Japanese re-analysis
-merra1.25	NASA MERRA Reananlysis reduced grid: [89.375S,89.375N], [179.375W,179.375E]
-merra0.5	NASA MERRA Re-analysis full grids: [90S,90N], [180W,180E)
-geos0.25 **-geos0.5** **-geos1x125** **-geos1x1** **-geos4x5** **-geos2x25**	NASA GEOS grids: [90S,90N], [180W,180E)
-fv1x125 **-fv2x25** **-fv4x5**	NASA Finite-volume GCM (a.k.a. GEOS-4): [90S,90N], [0,360)

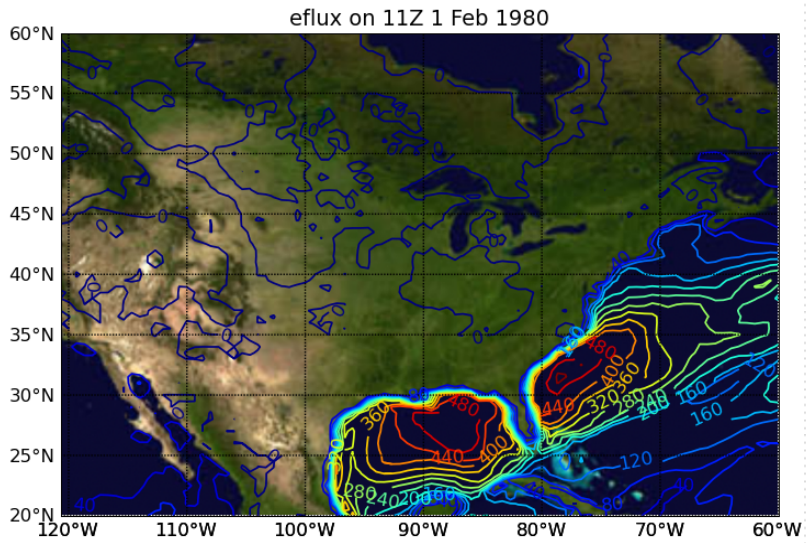
Examples with MERRA Data

Ex001: Simple Subsetting

Read the MAT1NXFLX collection (hourly surface fluxes) and create a local NetCDF file with 3 variables, for a region around North America, for a single day:

```
% lats4d.sh -dods \
  -i http://goldsmr2.sci.gsfc.nasa.gov:80/dods/MAT1NXFLX \
  -vars eflux hflux pblh -time 0z1feb1980 23z1feb1980 \
  -lat 20 60 -lon -120 -60 -o ex001 -v
```

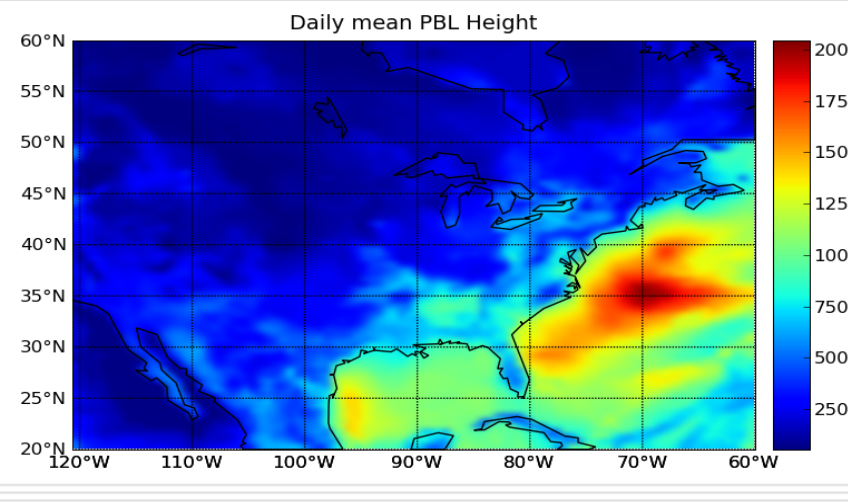


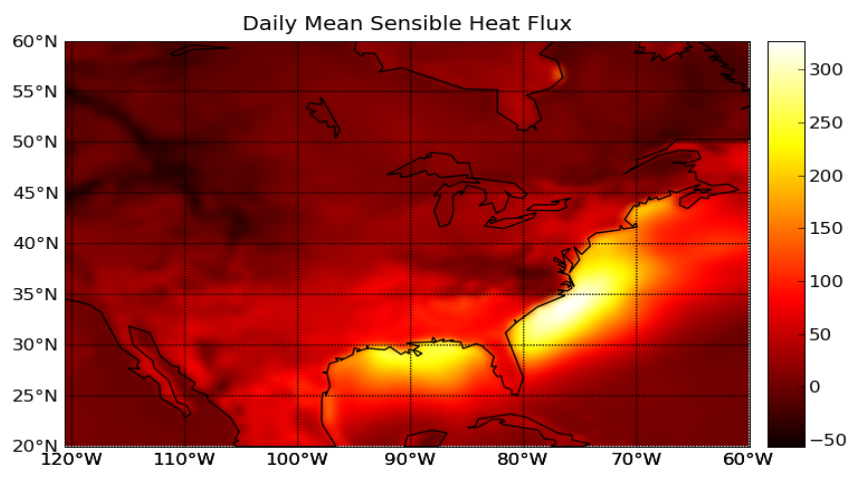


Ex002: Time mean

Read the MAT1NXFLX collection (hourly surface fluxes) and create a local NetCDF file with the daily mean of 3 variables, for a region around North America, for a single day:

```
% lats4d.sh -dods \
  -i http://goldsmr2.sci.gsfc.nasa.gov:80/dods/MAT1NXFLX \
  -vars eflux hflux pblh -time 0z1feb1980 23z1feb1980 \
  -lat 20 60 -lon -120 -60 -mean -o ex001 -v
```

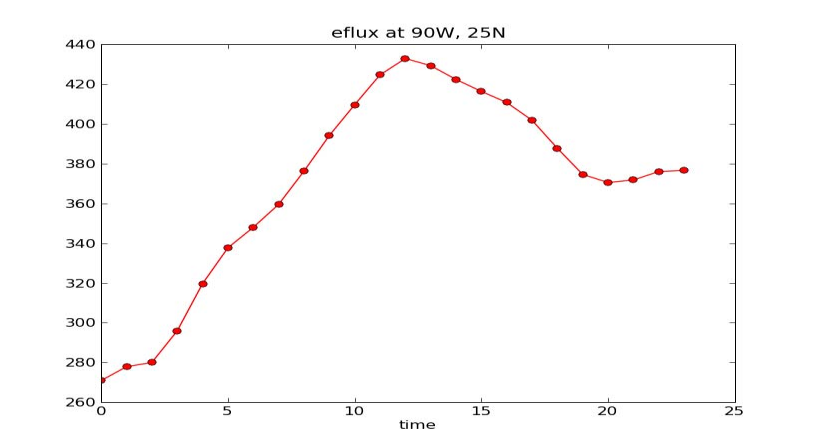


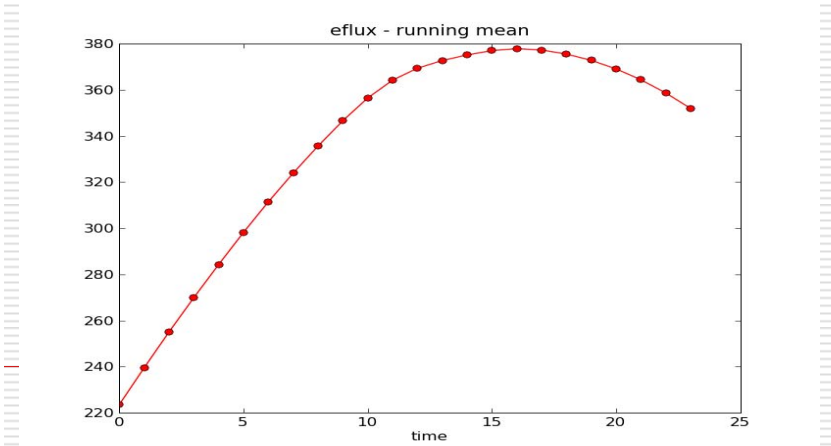


Ex003: Running mean

Read the MAT1NXFLX collection (hourly surface fluxes) and create a local NetCDF file with the daily running mean of a single variable, for a region around North America, for a single day:

```
% lats4d.sh -dods \
  -i http://goldsmr2.sci.gsfc.nasa.gov:80/dods/MAT1NXFLX \
  -vars eflux -time 0z1feb1980 23z1feb1980 \
  -lat 20 60 -lon -120 -60 -func 'ave(@,t-12,t+12)' -o ex003 -v
```





Ex004: Basic Statistics

Read the MAT1NXFLX collection (hourly surface fluxes) and creates ASCII output with basic statistics about 3 variables, for a region around North America, for a single day:

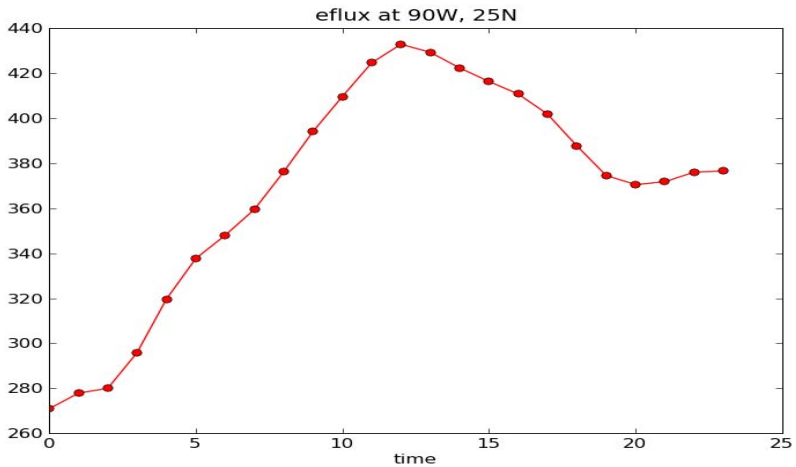
```
% lats4d.sh -dods \
  -i http://goldsmr2.sci.gsfc.nasa.gov:80/dods/MAT1NXFLX \
  -vars eflux hflux pblh -time 0z1feb1980 23z1feb1980 \
  -lat 20 60 -lon -120 -60 -format stats -v
```

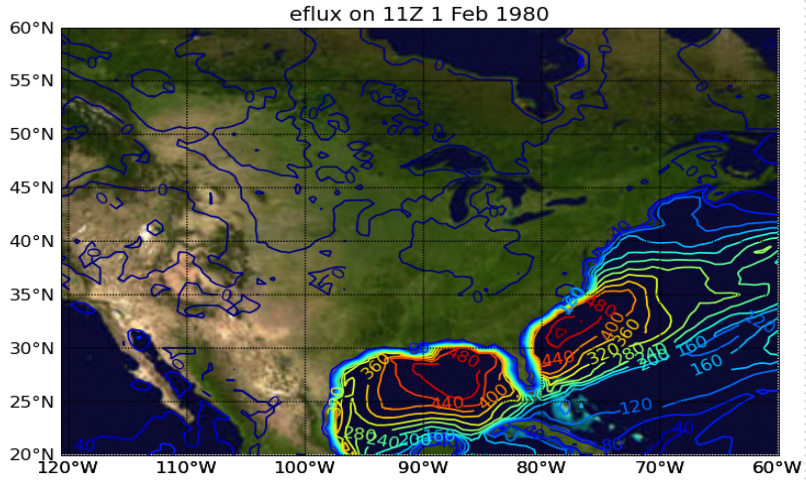
```
<stats>
+
+ <> Statistics on 00:30Z01FEB1980 for "MAT1NXFLX"
+
+    Name    Lev          Min          Max         MEAN          STDV          RMS
+ --------- ---- ------------ ------------ ------------ ------------ ------------
+ eflux      sfc    -12.7613     515.9260      45.1061      90.3556     100.9830
+ hflux      sfc    -96.6073     297.6430       7.0679      48.0070      48.5214
+ pblh       sfc     53.3895    2177.7300     410.4600     442.7710     603.7360
+
+ <> Statistics on 01:30Z01FEB1980 for "MAT1NXFLX"
+
+    Name    Lev          Min          Max         MEAN          STDV          RMS
+ --------- ---- ------------ ------------ ------------ ------------ ------------
+ eflux      sfc    -12.6806     512.3820      46.5060      93.9539     104.8280
+ hflux      sfc    -96.7478     299.0020       7.9609      49.4973      50.1301
+ pblh       sfc     53.3979    2261.7400     397.4950     447.2900     598.3680
+
Etc…
```

Ex005: Simple Reformatting

Read the MAT1NXFLX collection (hourly surface fluxes) and create a local GRIB-1 file with 3 variables, for a region around North America, for a single day:

```
% lats4d.sh -dods \
  -i http://goldsmr2.sci.gsfc.nasa.gov:80/dods/MAT1NXFLX \
  -vars eflux hflux pblh -time 0z1feb1980 23z1feb1980 \
  -lat 20 60 -lon -120 -60 -format grads_grib -o ex005 -v
```





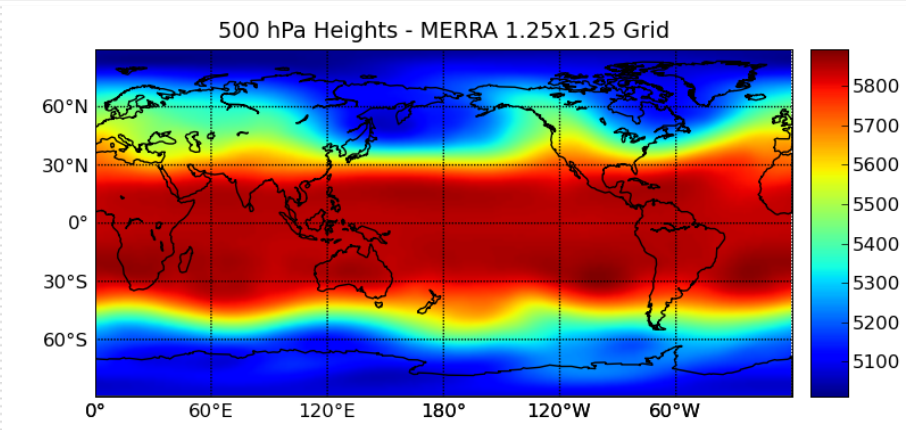
Horizontal Regridding

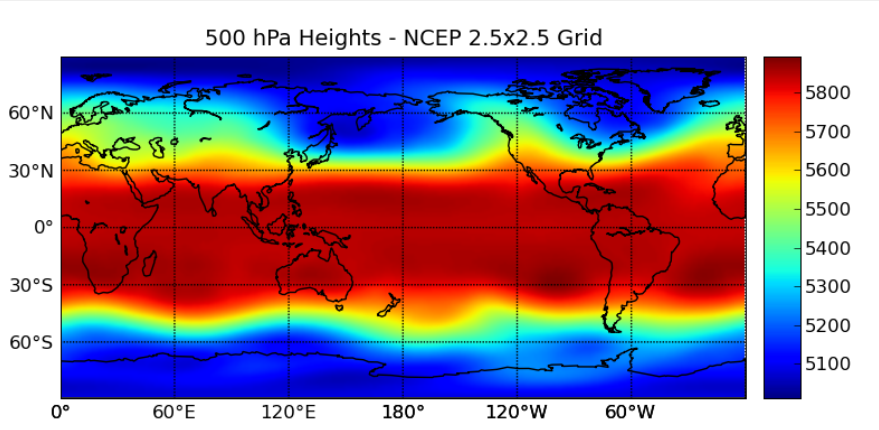
- LATS4D can apply any GrADS function to each variable as they are being written to a file. In particular the **`re()`** regridding function.
  - Caution: **`re()`** is not yet available in COLA's release of GrADS
  - Available in most OpenGrADS releases
    - v1.9-rc1, v2.0.a4.oga.1
    - Your USB memory stick has it!
- Documentation is at http://opengrads.org/doc/udxt/re/

Ex006: Simple Regridding

Read the MAIMCPASM collection (monthly mean assimilation) and create a local NetCDF file with 1 variable at a single level, globally, for a single day:

```
% lats4d.sh -dods \
  -i http://goldsmr3.sci.gsfc.nasa.gov:80/dods/MAIMCPASM \
  -vars h -levs 500 -time feb1980 feb1980 \
  -ncep2.5 -o ex006 -v
```





Impersonating another reanalysis

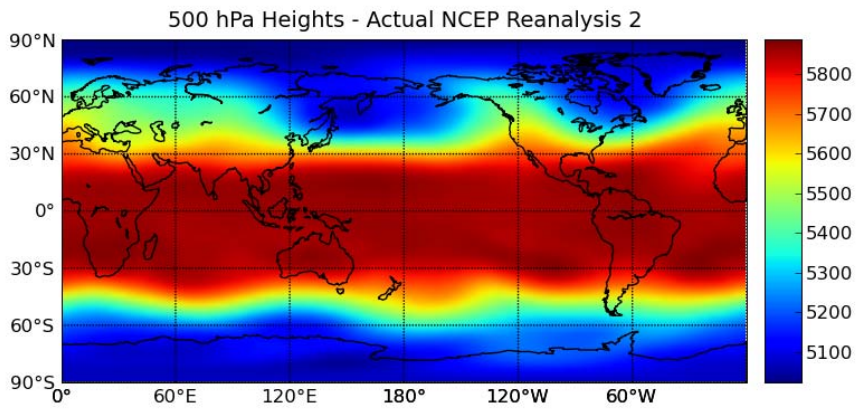
- The regridding of the previous example produced a MERRA file on he same grid as the NCEP Reanalysis 2
  - However, variable names were still as in MERRA
- LATS4D cannot (yet) rename variables on the fly
- The work around is to create a GrADS control file, alias the variable names, and then use lats4d for the regridding
  - Hint: use utility **make_ctl.sh**

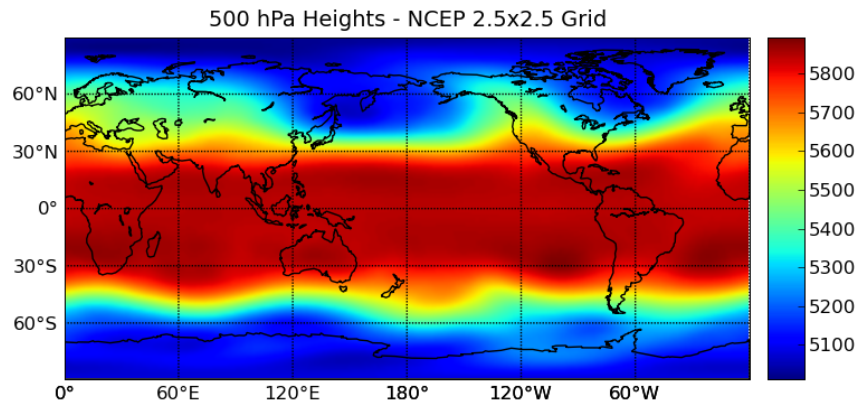
```
dset http://goldsmr3.sci.gsfc.nasa.gov:80/dods/MAIMCPASM
title Collection MAIMCPASM with NCEP R2 variable names
undef 1e+15
dtype netcdf
xdef 288 linear -180 1.25
ydef 144 linear -89.375 1.25
zdef 42 levels 1000 975 950 925 900 875 850 825
 800 775 750 725 700 650 600 550 500 450
 400 350 300 250 200 150 100 70 50 40
 30 20 10 7 5 4 3 2 1 0.7
 0.5 0.4 0.3 0.1
tdef 396 linear 00Z01JAN1979 1mo
vars 9
slp=>presmsl     0 t,y,x sea-level pressure
ps=>pressfc      0 t,y,x surface Pressure [Pa]
phis=>hgtsfc     0 t,y,x Geopotential height[gpm]
h=>hgtprs       42 t,z,y,x Geopotential height [gpm]
rh=>rhprs       42 t,z,y,x Relative humidity [%]
t=>tmpprs       42 t,z,y,x  Temperature [K]
u=>ugrdprs      42 t,z,y,x   u wind [m/s]
v=>vgrdprs      42 t,z,y,x   v wind [m/s]
omega=>vvelprs  42 t,z,y,x Pressure vertical velocity [Pa/s]
endvars
```

Ex007: Regridding with variable renaming

Read local file MAIMCPASM_as_NCEP-R2.ctl (which renames variables in the monthly mean assimilation collection) and create a local NetCDF file with 1 variable at 4 levels, globally, for a single day:

```
% lats4d.sh -dods \
  -i MAIMCPASM_as_NCEP-R2.ctl \
  -vars hgtprs -levs 850 700 500 300 -time feb1980 feb1980 \
  -ncep2.5 -o ex007 -v
```





Ex008:Comparing 2 reanalysis

Read local file MAIMCPASM_as_NCEP-R2.ctl (which renames variables in the monthly mean assimilation collection) and create a local NetCDF file with 1 variable at 4 levels, globally, for a single day:

```
% lats4d.sh -dods \
  -i ex007.nc  \
  -j http://nomad2.ncep.noaa.gov:9090/dods/reanalyses/reanalysis-2/month/pgb/pgb \
  -vars hgtprs -levs 850 700 500 300 -time feb1980 feb1980 \
  -format stats -v
```

```
+ <> Statistics on 00Z01FEB1980 for "ex007.nc"
+ <> Secondary input file is        "pgb"
 >>> Applying function "@.1-@.2"
+
+    Name    Lev          Min          Max         MEAN         STDV          RMS
+ --------- ---- ----------- ----------- ----------- ----------- -----------
  hgtprs     850    -32.7251     83.5547     -4.7193      6.1415      7.7450
  hgtprs     700    -39.6970     88.4153     -5.1268      6.8595      8.5634
  hgtprs     500    -47.4004     33.7495     -5.2056      9.4901     10.8237
  hgtprs     300    -64.9082     40.9922    -12.7557     11.4101     17.1139
  --------- ---- ----------- ----------- ----------- ----------- -----------
+ hgtprs       4    -64.9082     88.4153     -6.9519      8.2681     11.6550
```

Remarks

- The LATS4D examples we have just seen involves creating local files
  - This also provides a simple strategy for using MERRA data by applications that are not OPeNDAP enabled and canno read HDF-4
  - It also provides a method for creating NetCDF, GRIB and binary files
    - Recall that NetCDF files can be read jut fine by the HDF-4 library
- By using the GrADS interface to python, perl, PHP, TCL, Matlab, etc, these operations could also be perform on the fly without the need for an intermediate file.

Server-side regridding: Accessors

- These regridding operations/variable renaming can also be implemented server-side, on the fly
  - No need for creating a second copy of the dataset
  - By using the PDEF feature of GrADS all one needs to create is a new set of “ctl” for the new interface
  - This in turn can provide several *accessors* for MERRA data for easy comparison with other reanalysis

Concluding Remarks

- We have shown how subsetting, reformating and regridding can be performed client-side, on-demand
- OPeNDAP throughput may be an issue for performing such operations on large chunks of data
  - Implementing these operations server-side may alleviate these issues.